

# Komunikacja człowiek komputer - sprawozdania z aplikacji Mask Detector

Gorgoń Adam - 145278  
Grochowska Paulina - 145284

4 grudnia 2021

## 1 Wstęp

## 2 Prezentacja aplikacji

## 3 Dane

## 4 Użyte modele

Do wykrywania twarzy użyliśmy wytrenowanego modelu z biblioteki [OpenCV](#) w frameworku Caffe[2]. Do klasyfikacji, czy osoba posiada maskę stworzyliśmy własną sieć konwolucyjną zbudowaną na technologii tensorflow[1]. Model trenował się na cpu 3 godziny. Wczytywał zdjęcia z wcześniej wspomnianej bazy. Proporcje zbioru treningowego do testowego, wynosiły 0.8/0.2. Do lepszego wytrenowania model generował sobie zdjęcia, obracając zdjęcia o maksymalnie 15 stopni, robiąc lustrzane odbicie oraz przesuwając w pionie lub poziomie o 0.1 zdjęcia.

## 5 Działanie programu

### 5.1 Przygotowanie zdjęcia do modelu

Wczytane zdjęcie jest zmieniane na tablicę jako RGB. Następnie układ jest zmieniany na BGR. Wynika to z faktu, że podczas trenowania dane były wczytywane



(a) Zdjęcie bez obróbki z źle wczytanymi kolorami (b) Zdjęcie bez obróbki z dobrze wczytanymi kolorami

Rysunek 1: Wczytywanie zdjęcia

za pomocą biblioteki [OpenCV](#)(wczytuje bgr), a w aplikacji przez [Pillow](#)(wczytuje rgb).

Następnie rozjaśniamy obraz za pomocą korekcji gamma, by zmniejszyć efekt cienia na zdjęciu.

## 5.2 Detekcja twarzy

Model sprawdza na zdjęciu czy znajdują się twarz. Jeśli pewność algorytmu na to, że w przeszukiwanym w tym momencie prostokącie znajduje się twarz, wynosi ponad 0.5, to program przechodzi do klasyfikacji.

## 5.3 Klasyfikacja twarzy

Jeśli program znajdzie twarz, to "wycina"prostokąt uznany za twarz, zmienia jego rozmiar na 124x124 i wysyła do sieci konwolucyjnej. Sieć wyrzuca liczbę w zakresie  $[0, 1]$ . Jeśli liczba jest mniejsza od 0.5, to twarz ma na sobie założoną maskę, w przeciwnym razie algorytm uznaje, że na twarzy nie ma maseczki.

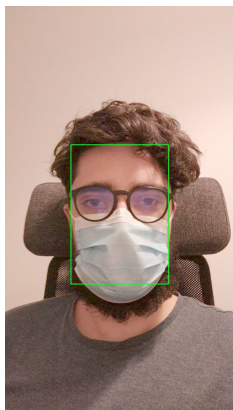


(a) Przed

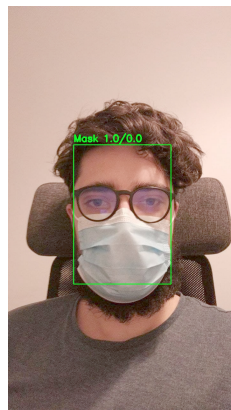


(b) Po

Rysunek 2: Korekcja gamma



(a) Algorytm znalazł twarz



(b) Algorytm sklasyfikował twarz

Rysunek 3: Działanie modelu

## 6 Podsumowanie

### Literatura

- [1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems.
- [2] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional Architecture for Fast Feature Embedding.